SSD: Single Shot MultiBox Detector

Abstract

- 異なる解像度の出力 map への射影により，1つのネットワークで異なる scale, aspect 比の prediction
- simple だから早い，高い accuracy

Introduction

- 多くのアプローチは bounding box を予測し，各 box を classifer の入力とする
  - Faster R-CNN の派生系
  - 計算量多すぎてリアルタイム厳しい
  - 高速化は性能とのトレードオフだった
- 提案手法では pixel を resample しない最初の deep network based なシステムにより，精度を維持しつつ大幅な速度の向上
- We summarize our contribution as follows:
  - YOLO よりも早く，遅く高精度な Faster R-CNN 並みに高精度な SSD の提案
  - feature maps に適用された小さな conv filter を用いた fixed set of default bounding box によって score と box offsets を予測する
  - 複数の scale の feature maps から予測することにより high accuracy を達成，prediction と aspect ratio を明確に分離した？
  - これらの設計はシンプルな end-to-end training を可能にし，速度と精度のトレードオフを大きく改善

2 The Single Shot Detector (SSD)

- 2.1 SSD framework for detection
- 2.2 training について
- 2.3 dateset-specific model details and experimental results

2.1 Model

- SSD は fixed-size collection of bounding boxes とクラスのスコアを出力する convs と，それに接続し最終的な detection を行う non-maximum suppression からなる
- base network は VGG-16
  - それに以下に示すような detection 用の構造を追加

Multi-scale feature maps for detection

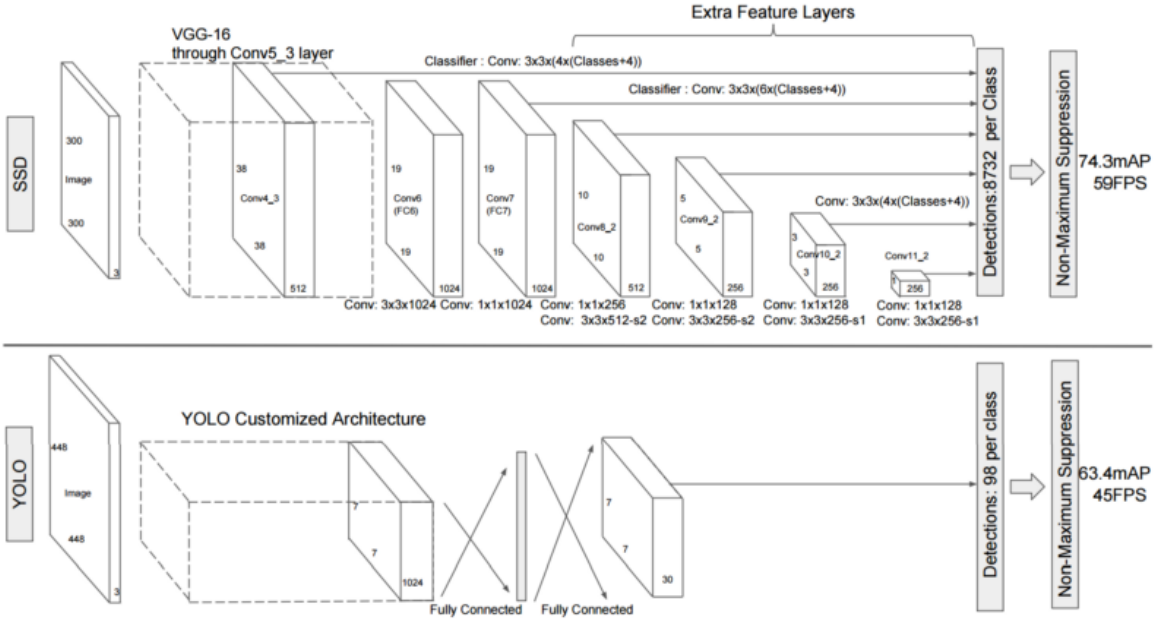
- truncated base network に conv feature layers を追加
  - だんだんサイズが小さくなるので multi scale な prediction が可能

Convolutional predictors for detection

- 各 added feature layer は detection prediction を conv filters を用いて出力する
- $m \times n$ feature map 上の各 cell について，bounding box の offset と各クラスのスコアを出力

Default boxes and aspect ratios

- $m \times n$ feature map 上の各 cell について，k 個の default box が存在し，それぞれが x, y, w, h と各クラスのスコアを出力する
  - したがって，$m \times n$ feature map のoutput は $(c + 4)kmn$ 個となる
  - Faster R-CNN の anchor box に似ているが，複数の feature maps に適用している点が異なる




```
print(4*(38**2+3**2+1**4)+6*(19**2+10**2+5**2))
```

```
8732
```

2.2 Training

- SSD と typical な region proposal を使う detector との重要な違いは，ground truth information を出力のうち特定のものに assign すること
  - YOLO や Faster R-CNN の RPN でもやってる
  - end-to-end で学習できる
- training は default boxes とその scales の選択，及び hard negative mining と data augmentation も含む

Matching strategy

- ground truth と default box を対応づける必要がある
  - まず，各 ground truth について jaccard overlap (MultiBox) が一番よく一致している default box を選択
  - 次に MultiBox と異なり，各 default box について jaccard overlap が threshold (0.5) より大きい ground truth を全て選択
  - 最大の overlap を持つ1つのみの default box を選ぶよりも複数の box を選ぶ方が learning problem が簡単になる

Training objective

- SSD の目的関数は MultiBox から派生しているが，複数カテゴリの object を扱えるように拡張してある
- x_{ij}^p: indicator for matchng i-th default box to the j-th ground truth box of category p
  - 上述の matching strategy の通り，$\sum_i x_{ij}^p \geq 1$
- objective loss function:

$$L(x, c, l, g) = \frac{1}{N}\left(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)\right)$$

  - N: number of matched default boxes ($L = 0$ when $N = 0$)
  - x: input
  - c: multiple classes confidences (c_i^p は i 番目の default box が p 番目のクラスに属するという confidence)
  - l: predicted box
  - g: ground truth
- localization loss ($L_{loc}(x, l, g)$): predicted box と ground truth の SmoothL1 loss
- box の中心 (cx, cy) と w, h を default bounding box (b) に回帰
  - Pos: match した default box の集合 (N 個)

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \tag{1}$$

$$\begin{gathered} L_{loc}(x, l, g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^{k} \text{smooth}_{\text{L1}}(l_i^m - \hat{g}_j^m) \\ \hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^w \qquad \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^h \\ \hat{g}_j^w = \log\left(\frac{g_j^w}{d_i^w}\right) \qquad \hat{g}_j^h = \log\left(\frac{g_j^h}{d_i^h}\right) \end{gathered} \tag{2}$$

$$L_{conf}(x, c) = -\sum_{i \in Pos}^{N} x_{ij}^p log(\hat{c}_i^p) - \sum_{i \in Neg} log(\hat{c}_i^0) \quad \text{where} \quad \hat{c}_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)} \tag{3}$$

Choosing scales and aspect ratios for default boxes

- 多様な scale のオブジェクトに対応するために，複数サイズの画像を入力し，出力をまとめるみたいなことがされてきた
  - SSD では feature maps の導入により parameters (convs だよね) を共有しながら同様の効果を得ることができている
- 低解像度，高解像度両方の feature map を利用した
- 特定の feature map が学習する object の大きさを規定した
- scale について
  - $s_k : x, y, w, h$ を画像 (feature map) 全体の辺の長さを1としたときの比として表現
  - $s_{\min} = 0.2$: lowest layer has a scale of 0.2
    - 一番解像度高いやつって理解で合ってる？たかだか5分割なのは嘘くさくない？
  - $s_{\max} = 0.9$: highest layer has a scale of 0.9
    - 一番解像度低いやつって理解で合ってる？てか画像全体だよね
  - m 枚の feature map

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m - 1}(k - 1), \quad k \in [1, m]$$

- aspect 比について
  - $a_r \in \{1, 2, 3, \frac{1}{2}, \frac{1}{3}\}$
- w, h の決め方

$$w_k^a = s_k\sqrt{a_r}, \; h_k^a = s_k/\sqrt{a_r}$$

  - ただし，$a_r = 1$ のときだけは $s_k' = \sqrt{s_k s_{k+1}}$ を追加
- x, y の決め方
  - feature map の各セルの中心
  - 論文中の式は $|f_k| \times |f_k|$ のセルについて書かれてる

Hard negative mining

- 特に default box が多い時は大半が negative
- pos, neg 間の imbalance をもたらす
- 全ての neg を使うのではなく，各 default box について confidence loss (L_{conf}) が高いものを採用し，neg:pos = 3:1 になるようにした

Data augmentation

- patch
  - image 全体を使う
  - minimum jaccard overlap with the object が 0.1, 0.3, 0.5, 0.7, or 0.9 になるように patch とる
  - randomly sample a patch
- patched image は元画像の 0.1 ~ 1 倍
- アスペクト比は2倍以下
- ground truth box の中心が sampled patch 内にある時は overlapped part を keep？
- 上述のことをやってから画像サイズを修正，さらに確率 0.5 でhorizontally flipped

- [14] と同様な photo-meric distoritions

Experimental Results

Base network

- ILSVRC CLS_LOC VGG16 を pre-train
  - [17] のように，fc6, fc7 を conv に
  - subsample parameters from fc6 and fc7 (パラメータ引き継いだのかな？)
  - change pool 5 from 2 x 2 - s2 to 3 x 3 - s1
  - use the a trous algorithm [18] to fill the 'hole'
  - remove all the dropout layers and fc8

3.1 PASCAL VOC2007

- VOC2007 (4952 images) について Fast R-CNN, Faster R-CNN と比較
- 新しく追加した layer のパラメータは 'xavier' method [20] により初期化
- conv4_3, conv10_2, conv11_2 については $a_r = 3, 1/3$ を除外し 4 default boxes
- conv4_3 の feature scale は違うので， L2 normalization した
- SSD512 では convs12_2 を追加し，$s_{\min} = 0.15$とし conv4_3 は $s = 0.07$
- COCO で train した SSD512 を 07+12 で fine-tune するのが最強，81.6 %
- Fig. 3 に detection analysis tool [21] による結果
  - 横軸の total detections の意味がわからん，[21] 読まないとかな…
  - recall が高い
    - weak criteria (0.1 jaccard overlap) の方が高い
  - R-CNN に比べ localization error が低い
    - end-to-end なシステムだからかな
  - 似たようなカテゴリ，特に動物の識別に弱い
    - partly because we share locations for multiple categories???
  - box のサイズに敏感，特に小さいのに弱い
    - higher layer だと情報消えてるし当たり前だよね，input size あげればましになるけど改善の余地あり

3.2 Model analysis

- YOLO に似てる data augmentation した，これがむっちゃ効いた
- atrous 使うと結果は同じだけど 20% くらい学習早い
  - parameter の初期値の話かな
- multiple output layers at different resolutions is better

3.3 PASCAL VOC2012

- COCO で train した SSD512 を 07+12 で fine-tune するのが最強，80.0 %

3.4 COCO

- COCO は小さい object が多いので default box の scale を小さくした (0.2 -> 0.15)
  - SSD300 では conv4_3 は $s = 0.07$
    - conv4_3 はもともと幾つなんや，$s_{\min}$ は feature maps の話っぽいよね
  - SSD512 では convs12_2 を追加し，$s_{\min} = 0.1$とし conv4_3 は $s = 0.04$

3.5 Preliminary ILSVRC results

- SSD300 で 43.4 mAP...

3.6 Data Augmentation for Smaller Object Accuracy

- 小さい object の検出は SSD にとって難しい，data augmentation でなんとかしたい
- 2.2 で述べた random crop は 'zoom in' に相当する，小さい object について train するためには 'zoom out' もしたい
- 平均画素値で満たされた元画像の 16 倍の大きさの canvas 上に random に画像を置いてから，random crop operation した？？
    - 無駄にデカすぎるきがするけど，object が 0.1 とか (jaccard overlap) あればいいからこれくらい必要なのかな？
    - ちょっとこれ解釈怪しいかも，実装見ないとかな…
- 2%-3% 上がった，これもやろう